Arcnet Support

Implementation for Local Station

Jun 10, 1991

Introduction

Arcnet is supported by the local station system software to permit access to Smart Rack Monitors (SRMs) which are used in the new Linac controls. As an arcnet node, an SRM interfaces to 64 A/D channels, 16 D/A channels, 8 bytes of binary I/O, and additional I/O depending upon the installation of daughter boards. Onboard an SRM is a 68332 processor, which is used both for data collection and arcnet communication.

The general plan for networking to SRMs is to use Acnet-header-based network messages, which support generic task-task communication over a network. A local station connects to a few (1–8) SRM arcnet nodes. Every 15 Hz cycle, the local station requests all data from each SRM to be collected and returned in a single frame. (In this way, the SRM does not need a 15 Hz interrupt signal to announce the start of a new cycle.) The returned data is mapped into local station analog channel readings and binary status byte readings. Use of a broadcast request permits multiple SRMs to simultaneously prepare their data for arcnet transmission. Each SRM returns a single frame of data containing readings of all types. The token-passing arcnet network hardware arbitrates frame delivery.

SRM-based data acquisition is designed to collect data more efficiently than the previous scheme (used heavily in the D0 system) of using 1553-based Rack Monitors. The overhead of processing a single arcnet frame replaces the overhead involved with processing the many 1553 commands to collect data from the analog and binary hardware interfaces in the D0 Rack Monitors. The inclusion of a processor onboard the SRM allows consideration of implementing certain special handling such as closed loop logic in the SRM rather than in the local station.

Network Layer

As used in the local station, the Network Layer refers to support of Acnet-header-based network messages. Providing task-task communication across a network, it provides a higher level interface for applications that run in the local station. A lower level protocol may have been sufficient for use with SRMs, but the Network Layer permits arcnet communication to take advantage of more software that already exists.

Transmission

The current network routines do not include an argument to specify which network is being used, so the use of Arcnet communications is based upon a certain range of destination node#s. Arcnet node#s are of the form $7Axx. For reasons having to do with setting support for the SRMs, the range of xx is

A message to be sent to a network is placed into an allocated message block, and the routine OUTPQX is called to queue the message to the network. It places the message block pointer into a network output queue. With token ring, this is the OUTPQ system table. With Arcnet, it is a new table statically allocated in on-board ram. (A network output queue need not be in non-volatile memory.) The OUTPQX routine checks the destination node# to determine which queue to use.

When it is time to flush queued messages to the network hardware, the routine NetXmit is called. It now accepts a parameter that indicates which network output queue is to be flushed. (The Network Layer routine NetSend calls NetXmit once for each network.) It concatenates consecutive messages destined for the same node into frames according to the maximum frame size for that network. Most of the NetXmit logic is independent of the network, but a few local variables are set that depend upon the network being used. They are the network# (0 or 1), network board address, network output pointer queue, TPL header, and the maximum frame size. The TPL header is used to emulate the token ring Transmit Parameter List chain that contains references to the spooled frames to be processed by the network hardware. Transmit interrupt processing sequences to the next frame waiting in the TPL chain.

At the end of the NetXmit routine, when a frame is ready to be sent to the hardware a special routine is called that depends on the network being used. This routine ensures that the frame just placed into the TPL chain will be ultimately sent out by the hardware. If the network is not already busy, it forces a transmit interrupt to get it started. If it is busy, the transmit interrupt that results from completion of the current frame will do the job as it follows the TPL chain.

Data structures for arcnet

Several data structures are significant for arcnet communications. The ARCPQ is the arcnet output pointer queue for messages described above. The ARCXMTB is the circular buffer used to hold prepared arcnet frames which are referenced by entries in the ARCTPLH transmit parameter list chain. The ARCRCVB is a circular buffer into which received frames are copied from the hardware buffers. The SRMTABL is used for communications with SRMs specifically, especially for data acquisition. A set of arcnet variables is maintained in the TRING system table, which holds some common data structures with token ring. Note that a special word at TRING+$32 must be set to ‘AR’ to enable a local station’s use of arcnet at all; *without that key, the system will not even look for the arcnet hardware board,* the CC121 VME module made by CompControl.

Data acquisition

Arcnet data acquisition from SRMs is directed by entries placed in the data received data into local station analog channel and binary byte readings.

Since all data acquisition is processed while the Update Task is active, in order to preserve correlated data, a special check is made by the arcnet receive interrupt routine for data acquisition replies to direct them to the appropriate message queue. (This is normally done by the ANET Task, which handles acnet-header-based frames and routes each message to the appropriate message queue, but the ANET Task cannot run until the Update Task has finished. Since data acquisition from SRMs is the only reason for supporting arcnet, it was decided that this special case handling was justified.) Data acquisition replies are detected based upon the source task id (in the acnet header) that identifies the network connection used by the Update Task for making such requests.

SRM support

Details of the support for SRM data requests and settings, including the formats for the data access table entries and device setting parameter specifications, are found in the document called “SRM Message Protocols.”